

Amendments to the Specification:

The paragraph beginning at line 29 on page 10, ending on page 11, line 11:

In addition to being coupled to the BS **14**, the terminal **10** can be wirelessly coupled to one or more wireless access points (APs) **30**. The APs can comprise access points configured to communicate in accordance with techniques such as, for example, radio frequency (RF), Bluetooth (BT), infrared (IrDA) or any of a number of different wireless networking techniques, including WLAN techniques. The APs may be coupled to the Internet **22**. Like with the MSC **16**, the APs can be directly coupled to the Internet. In one advantageous embodiment, however, the APs are indirectly coupled to the Internet via a WAP GTW, including a push proxy/GTW and/or method proxy/GTW. As will be appreciated, by directly or indirectly connecting the terminals and the push initiator **24**, origin server **26**, as well as any of a number of other devices, to the Internet, the terminals can communicate with the push initiator, origin server, etc., and with one another to thereby carry out various functions of the terminal, such as to transmit data, content or the like to, and/or receive content, data or the like from, the push initiator, origin server, etc.

The paragraph beginning at line 1 and ending at line 11 on page 15:

It is understood that the controller **48** includes the circuitry required for implementing the audio and logic functions of the mobile station. For example, the controller may be comprised of a digital signal processor device, a microprocessor device, and various analog to digital converters, digital to analog converters, and other support circuits. The control and signal processing functions of the mobile station are allocated between these devices according to their respective capabilities. The controller thus also includes the functionality to convolutionally encode and interleave message and data prior to modulation and transmission. The controller can additionally include an internal voice coder (VC) **48A**, and may include an internal data modem (DM) **48B**. Further, the controller may include the ~~functionally~~ functionality to operate one or more software applications, which may be stored in memory.

The paragraph beginning at line 12 and ending at line 23 on page 18:

Upon receipt of the subscription request, the push initiator **24** can respond by sending a public key to the terminal **10**, where the public key is associated with the subscription request. The terminal can then store the public key, such as in memory **42**, where the public key can be stored along with an association to the subscription request (such as in instances in which the terminal authorizes more than one push initiator). As will be appreciated by those skilled in the art, in addition to being associated with the subscription request (at the push initiator and/or terminal) the public key also has an associated private key. In this regard, the push initiator maintains the private key in a private manner, but makes the public key ~~(non-secret) key~~ (non-secret key) available to the terminal. As explained below, the push initiator can utilize the private key to digitally sign data transmitted to the terminal. The terminal can thereafter verify such data as originating from the authenticated push initiator based upon the digital signature and the public key.

The paragraph beginning at line 18 and ending at line 27 on page 19:

After the push initiator sends the digitally signed service loading content, the terminal **10** can pull the content identified by the URI upon receipt of the digitally signed service loading content, and without user interaction. Before pulling the content, however, the terminal can, again without user interaction, automatically authenticate the service loading content by verifying the digital signature with the service loading content. In this regard, the terminal can use the public key sent by the push initiator ~~sent to the terminal~~ to verify the digital signature. By verifying the digital signature, the terminal can verify that the service loading content originated at an authorized push initiator **24**, and can verify that the service loading content identifies authorized content at an authorized origin server **26**, if such were designated in the subscription request.

The paragraph beginning at line 9 and ending at line 19 on page 24:

During the download session, then, the download agent **76** can utilize the information packets to monitor the received data packets. More particularly, the download agent can monitor the received data packets to track the number and accuracy of the received packets, blocks of packet(s) or portion of the requested content. In this regard, the number of received packets, blocks of packet(s) or portion of the requested content can be compared to an identified number of packets ~~in or size~~ of the requested content at any one or more points during the download session. The accuracy of the received packets, blocks of packet(s) or portion of the requested content can be verified in any of a number of different manners, such as by utilizing CRC data in each of the packets. Similarly, a CRC or any other characteristic of accuracy can be included in each transmitted packet, block of packet(s) or portion of the requested content.

The paragraph beginning at line 20 on page 24 and ending at line 3 on page 25:

By tracking the number and accuracy of the received packets, the download agent **76** can identify if, and when, a download session has experienced an interruption. In this regard, an interruption in the download session can be identified in any of a number of different manners. For example, the download session can be regarded as interrupted if the total number of received packets, blocks of packet(s) or portion of the requested content, which can generally be identified by the download descriptor ~~72-82~~ and/or one or more information packets, does not equal the total number of received packets, blocks of packet(s) or size of the requested content when the download session ends or is terminated. Also, the downloaded session can be regarded as interrupted if the number of erroneously received packets exceeds a predetermined number of packets when the download session ends or is terminated, where the predetermined number of packets can be dependent on the type of the downloaded content (absolute number or relative) when the download session ends. In addition, the download session can be regarded as interrupted if a user of the terminal **10** terminates the download session, or if the download session is terminated by the terminal (e.g. due to power failure).

The paragraph beginning at line 25 on page 25, and ending at line 5 on page 26:

After determining how much of the download has been completed, the download agent 76 can be instructed to recover the interrupted packet, as well as the remaining packets to be sent to the terminal 10 to complete the download of the content. In this regard, the application can send an instruction, such as a resume download message, to the download agent to thereby instruct the download agent to recover the interrupted packet and download the remaining packets. In turn, the download agent can send an instruction, such as a resume download message, to the origin server. In this regard, the instruction can specify a session identifier identifying a download session to the origin server, an object identifier identifying the requested content, and/or a pointer to the interrupted packet. The instruction can also include an identifier identifying the terminal, and/or a description of a cause for the interruption.